C
B1
Con't

encrypting a block of input plaintext data utilizing said [object] key schedule; [in conjunction with an encryption process]

modifying the at least one object key;

modifying the key schedule based upon the at least one modified object key;

encrypting a next block of input plaintext data utilizing said modified key schedule; and

repeating the steps of modifying the at least one object key, modifying the key schedule and encrypting utilizing the modified key schedule until the encrypting of blocks of plaintext data is completed.

2. (Amended) A computer implemented method as defined in Claim 1, wherein the [encryption process comprises a block cipher system such that blocks of data bits are encrypted] modification of the key schedule is independent of the input plaintext data.

B2
Sub C2

5. (Amended) A computer implemented method as defined in Claim 1, further comprising prior to the first step of encrypting the steps of:

creating an initial state of the at least one object key by the user;

creating an initial state of a random session object key;

encrypting the initial state of the random session object key in a block cipher encryption process with the initial state of the at least one object key; and

~~B2~~C12 cont

wherein the step of modifying the at least one object key is based on seeding from the random session object key [before each input data block] so that each block of input plaintext is encrypted based on a different object key.

5. ~~7~~. (Amended) A computer implemented method as defined in Claim ~~6~~, 4 wherein a new random number is generated and assigned as the initial state of the random session object key for each block of input plaintext to be encrypted. [time the block cipher encryption process is executed].

B3

6. ~~8~~. (Amended) A computer implemented method as defined in Claim ~~5~~, 3 wherein the method comprises two [each] object [key is associated with a different] keys utilized to produce the key schedule for [encrypting] each block of input [data block with said different key schedule] plaintext.

7. ~~9~~. (Amended) A computer implemented method as defined in Claim [4] 1, wherein the method of modifying the [dynamic] at least one object key comprises the steps of:

generating a random seed unsigned byte and bit wise exclusive or to an unsigned byte of a current state of the at least one object key provided by an incremented index into the current state of the object key (I_BYTE_OBJECT_KEY);

performing an unsigned byte addition on the output byte of the previous operation (PREV_OUTPUT) with I_BYTE_OBJECT_KEY;

performing a 16-bit multiplication of PREV_OUTPUT and I_BYTE_OBJECT_KEY modulus 254 and add 2;

performing a 16-bit addition of PREV_OUTPUT and I_BYTE_OBJECT_KEY;

performing another 16-bit addition of PREV_OUTPUT and I_BYTE_OBJECT_KEY;

B3 cont

performing a bit-wise exclusive or of PREV_OUTPUT with a 16-bit unsigned integer of the current state of the object key provided by an incremented index into the current state of the object key (I_INT_OBJECT_KEY);

rotating PREV_OUTPUT to the right I_BYTE_OBJECT_KEY modulus 15 plus 1 times;

performing a 16-bit addition of PREV_OUTPUT and I_INT_OBJECT_KEY;

performing a 16-bit multiplication of PREV_OUTPUT and I_INT_OBJECT_KEY with the lower order byte of I_INT_OBJECT_KEY modulus 254 plus 2;

performing a 16-bit addition of PREV_OUTPUT and I_INT_OBJECT_KEY;

performing another 16-bit addition of PREV_OUTPUT and I_INT_OBJECT_KEY;

performing a bit-wise exclusive or of PREV_OUTPUT with a 32-bit unsigned long integer of the current state of the object key provided by an incremented index into the current state of the object key (I_LONG_INT_OBJECT_KEY);

rotating PREV_OUTPUT to the left I_BYET_OBJECT_KEY modulus 31 plus 1 times;

B3 cont

performing a bit-wise exclusive or of PREV_OUTPUT with I_LONG_INT_OBJECT_KEY;

repeating the previous set of operations eighty-four times substituting the random seed unsigned byte with a byte from a four byte output block provided by the previous set of operations recursively setting the current output block to a next output block when the current output block is exhausted, utilizing a different ordered byte each round;

performing a byte transposition of the bytes in the new 256 byte output block (N_OUTPUT) provided by the previous set of operations utilizing the following steps:

performing a byte-wise index through N_OUTPUT; switching the current byte of N_OUTPUT with the N-OUTPUT byte indexed at position I_BYTE_OBJECT_KEY; and indexing through the entire block of N_OUTPUT.

19. ~~21~~. (Amended) A computer implemented method as defined by Claim ~~2~~1, 2/13/01 SK wherein the block cipher encryption process comprises the steps of:

B4

transpositioning a substitution array whose elements contain unique numbers in reference to said substitution array by switching a position of each element with a position provided by an element of a key, which position provided by an element of the key is bounded by the size of said substitution array which is composed of 256 elements;

transpositioning a [traverse] transverse array whose elements contain unique numbers in reference to said transverse array by switching a position of each element with a position provided by an element of the key, the position provided by an element of the key is bounded by the size of the transverse array which is equal to the block size;

B4 cont

replacing each input byte transverse number of times with the value of the substitution array indexed with the input byte;

summing each output byte of the previous three steps to an element of the key to create ciphertext;

grouping the ciphertext in a 32-bit sliding window and rotating to the left an element of the key modulus 31 plus 1 times, the window sliding by one byte after each rotation and this step being performed on all ciphertext bytes:

performing a bit-wise exclusive or of each cipher text byte to an element of the key;

transpositioning the substitution [arry] array by switching a position of each element with a position provided by an element of the key, the position provided by an element of the key is bounded by a size of said substitution array;

tanspositioning the transverse array by switching a position of each element with a position provided by an element of the key, the position provided by an element of the key is bonded by a size of said transverse array;

replacing each input byte transverse number of time with a value of the substitution array indexed with an input byte;

B4 cont'd

transpositioning the ciphertext by switching a position of each ciphertext element with a position provided by an element of the key, the position provided by an element of the key is bounded by a size of the block;

repeating the previous seven steps four times with the key elements being unique each time the key is accessed;

transpositioning each bit in the ciphertext block by switching a position of each ciphertext bit with a position provided by elements of the key, the position provided by elements of the key are bounded by the size of the blocks times eight; and

repeating the previous nine steps four times with the key elements being unique each time the key is accessed.

20. ~~22~~. (Amended) A computer implemented method as defined in Claim ~~21~~ 19, wherein said key is the at least one object key.

Please add new Claims 36-40 as follows:

27.

--~~36~~. A computer implemented method for encrypting data comprising the steps of: creating at least one object key comprising data and methods that operate on said data; and

B5

modifying the at least one object key for each input block of plaintext utilizing the at least one object key in conjunction with an encryption process; wherein the step of modifying the at least one object key comprises the steps of:

generating a random seed unsigned byte and bit wise exclusive or to an unsigned byte of a current state of the object key provided by an incremented index into the current state of the object key (I_BYTE_OBJECT_KEY);

performing an unsigned byte addition on the output byte of the previous operation (PREV_OUTPUT) with I_BYTE_OBJECT_KEY;

performing a 16-bit multiplication of PREV_OUTPUT and I_BYTE_OBJECT_KEY modulus 254 and add 2;

performing a 16-bit addition of PREV_OUTPUT and I_BYTE_OBJECT_KEY;

performing another 16-bit addition of PREV_OUTPUT and I_BYTE_OBJECT_KEY;

performing a bit-wise exclusive or of PREV_OUTPUT with a 16-bit unsigned integer of the current state of the object key provided by an incremented index into the current state of the object key (I_INT_OBJECT_KEY);

rotating PREV_OUTPUT to the right I_BYTE_OBJECT_KEY modulus 15 plus 1 times;

performing a 16-bit addition of PREV_OUTPUT and I_INT_OBJECT_KEY;

B5 Cont

performing a 16-bit multiplication of PREV_OUTPUT and I_INT_OBJECT_KEY with the lower order byte of I_INT_OBJECT_KEY modulus 254 plus 2;

performing a 16-bit addition of PREV_OUTPUT and I_INT_OBJECT_KEY;

performing another 16-bit addition of PREV_OUTPUT and I_INT_OBJECT_KEY;

performing a bit-wise exclusive or of PREV_OUTPUT with a 32-bit unsigned long integer of the current state of the object key provided by an incremented index into the current state of the object key (I_LONG_INT_OBJECT_KEY);

rotating PREV_OUTPUT to the left I_BYET_OBJECT_KEY modulus 31 plus 1 times;

performing a bit-wise exclusive or of PREV_OUTPUT with I_LONG_INT_OBJECT_KEY;

repeating the previous set of operations eighty-four times substituting the random seed unsigned byte with a byte from a four byte output block provided by the previous set of operations recursively setting the current output block to a next output block when the current output block is exhausted, utilizing a different ordered byte each round;

performing a byte transposition of the bytes in the new 256 byte output block (N_OUTPUT) provided by the previous set of operations utilizing the following steps:

B5 cont

performing a byte-wise index through N_OUTPUT; switching the current byte of N_OUTPUT with the N-OUTPUT byte indexed at position I_BYTE_OBJECT_KEY; and indexing through the entire block of N_OUTPUT.

28. ~~37~~. A computer implemented method for encrypting data comprising the steps of:

creating at least one object key comprising data and methods that operate on said data; and

encrypting input plaintext data utilizing said object key in conjunction with a block cipher encryption process, wherein the block cipher encryption process comprises the steps of:

transpositioning a substitution array whose elements contain unique numbers in reference to said substitution array by switching a position of each element with a position provided by an element of a key, which position provided by an element of the key is bounded by the size of said substitution array which is composed of 256 elements;

transpositioning a traverse array whose elements contain unique numbers in reference to said transverse array by switching a position of each element with a position provided by an element of the key, the position provided by an element of the key is bounded by the size of the transverse array which is equal to the block size;

B5 cont'

replacing each input byte transverse number of times with the value of the substitution array indexed with the input byte;

summing each output byte of the previous three steps to an element of the key to create ciphertext;

grouping the ciphertext in a 32-bit sliding window and rotating to the left an element of the key modulus 31 plus 1 times, the window sliding by one byte after each rotation and this step being performed on all ciphertext bytes:

performing a bit-wise exclusive or of each cipher text byte to an element of the key;

transpositioning the substitution array by switching a position of each element with a position provided by an element of the key, the position provided by an element of the key is bounded by a size of said substitution array;

2/13/01 SK

transpositioning ~~tanspositioning~~ the transverse array by switching a position of each element with a position provided by an element of the key, the position provided by an element of the key is ~~bonded~~ bounded by a size of said transverse array;

2/13/01 SK

replacing each input byte transverse number of time with a value of the substitution array indexed with an input byte;

transpositioning the ciphertext by switching a position of each ciphertext element with a position provided by an element of the key, the position provided by an element of the key is bounded by a size of the block;

B5 cont

repeating the previous seven steps four times with the key elements being unique each time the key is accessed;

transpositioning each bit in the ciphertext block by switching a position of each ciphertext bit with a position provided by elements of the key, the position provided by elements of the key are bounded by the size of the blocks times eight; and

repeating the previous nine steps four times with the key elements being unique each time the key is accessed.

~~38~~ 29. A computer implemented method as defined in Claim ~~37~~ 28, wherein said key is the object key.

~~39~~ 30. A computer implemented method as defined in Claim ~~37~~ 28, wherein the last transpositioning step uses a switch key comprised of the following steps:

initializing the switch key with elements of an initial state of the object key;

grouping the switch key by 32-bit blocks;

replacing the current switch key element with the following process:

performing a bit-wise exclusive or of the current switch key element to a switch key element indexed two elements from the current element;

B5 Con't

rotating the output of the previous step to the right switch key indexed three elements from the current element modulus thirty-one plus one;

performing a bit-wise exclusive or of the output from the previous step to a switch key element indexed three elements from the current element;

repeating the previous three steps for each final transposition switch operation.

31. ~~40~~. A computer implemented method as defined in Claim 30 ~~39~~, wherein a hashing function is included in the creation of the switch key.--

REMARKS

At the outset, Applicant would like to thank the Examiner and his Supervisor for the courtesies extended their counsel during a telephone interview conducted on August 7, 2000. During the telephone interview, the essence of Applicant's invention was discussed as well as Applicant's understanding of the cited U.S. Patent No. 5,003,596 to Wood. Applicant has amended the claims in a sincere effort to more specifically define Applicant's invention in view of the prior art.

The Office Action refers to an earlier telephone discussion on March 6, 2000 related to a restriction requirement by the Examiner. During a telephone discussion, a provisional election was made without traverse to prosecute the invention of Group 1, Claims 1-24, 30-32